

ERPEEG Tutorial

Version 2.0

This tutorial was written by:

Sravya Atluri, Matthew Frehlich, Benjamin Schwartzmann and Dr. Faranak Farzan.

Contact: faranak.farzan@sfu.ca

Temerty Centre for Therapeutic Brain Stimulation
Centre for Addiction and Mental Health
Toronto, Canada

Table of Contents

Introduction	3
Installation	4
Dataset Loading and Setup	5
Settings.....	7
Step I – Initial Processing	9
1. Baseline Removal	9
2. Resampling.....	9
3. Channel location lookup	10
Step II – Removal of Bad Trials and Channels.....	11
1. Select Attribute for Display.....	11
2. Plot Trials.....	11
3. Plot Channels	13
4. EEGPLOT.....	15
5. Clear Subject	16
6. Deletion Matrix	16
Step III – Filtering	17
Step IV – ICA.....	18
Step V – Remove Artifacts with ICA.....	19
Step VI – Remove Channels and Trials 2.....	20
Step VII – Final Processing	20

Introduction

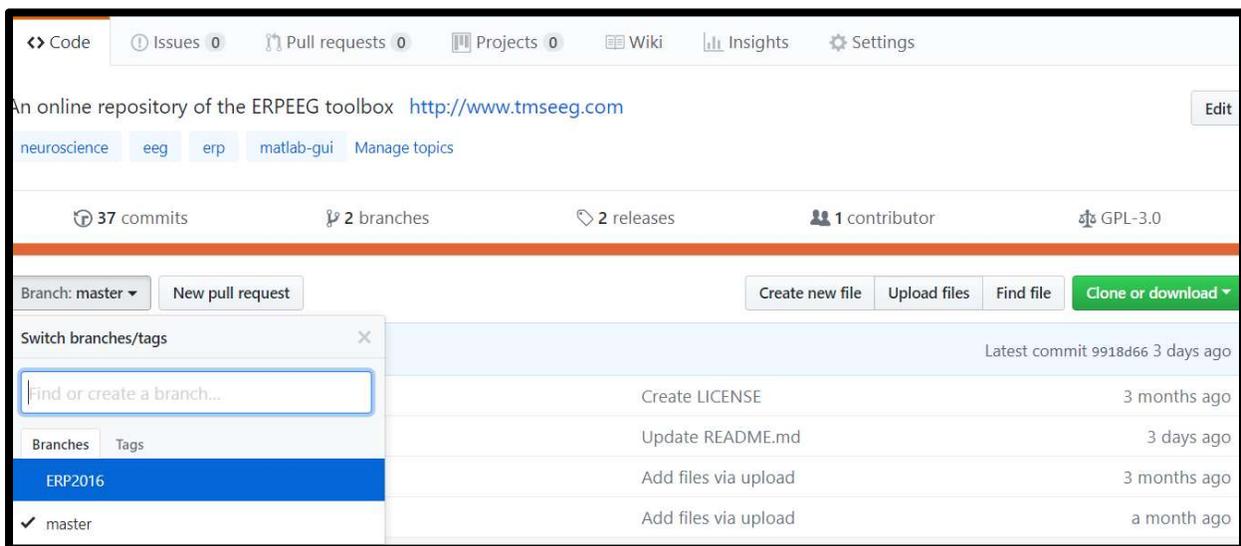
The ERPEEG toolbox aims to standardize and streamline the procedure for EEG data pre-processing (mainly resting-state EEG and event-related potential (ERP) data). A basic and generalized pipeline is provided to apply a number of artifact extraction techniques for targeted noise removal. The toolbox has been developed as an open-source MATLAB GUI application to provide an interactive and compact platform for data processing and artifact extraction. Some examples of artifacts that can be extracted using this toolbox include: muscle artifacts (EMG), eye movements or electrooculographic (EOG) artifacts, electrode movement artifacts, etc.

To describe in more detail, the ERPEEG toolbox contains several important features for standardized EEG data pre-processing. First, it incorporates interactive data visualization capabilities, allowing the user to visualize the data at each step of the workflow and verify the effectiveness of the data cleaning procedure. Second, intermediate datasets are saved after each processing step along with other important meta-information such as the deleted trials and channels, and the removed artifacts. This enables the user to easily revert to a previous step in the workflow, track their processing for consistency, and create a database of selected artifacts. Third, ERPEEG is a flexible platform. It allows for basic customization through the settings menu while providing a modular structure for advanced users to modify the order of processing or incorporate additional steps to accommodate processing of different EEG projects. Finally, parameters selected through the setting menu are saved in a separate MATLAB file. This enables future replication or assessment of the data pre-processing steps.

Installation

Before loading the electrophysiological dataset into ERPEEG, users must convert their dataset to .set format using the EEGLAB software suite (<http://sccn.ucsd.edu/eeglab/downloadtoolbox.php>). ERPEEG is highly dependent of EEGLAB for many of its functionalities and therefore the EEGLAB toolbox must be installed prior to data processing.

There are currently two versions of ERPEEG available. The last version works with MATLAB v2013 and higher (**MASTER BRANCH**) and can be found here (<https://github.com/EEGSignalProcessing/ERPEEG/>). We recommend using this last version but if needed, previous version (**ERPEEG2016 BRANCH**) can be found here (<https://github.com/EEGSignalProcessing/TMSEEG/tree/ERPEEG2016>).



ERPEEG requires installation of **EEGLAB v12.0.2.6b** or higher. It also requires:

- MATLAB signal processing toolkit. Make sure to set the MATLAB signal toolbox as a higher path priority than other software such as field trip, as some functions may share common names.
- FASTICA is the current preferred algorithm for ICA in ERPEEG
- More detail on the FASTICA algorithm can be found here: (<http://research.ics.aalto.fi/ica/fastica/code/dlcode.shtml>)

ERPEEG creates multiple intermediate datasets at each processing step to allow easy reprocessing of the data. We highly recommend that each base dataset for processing is placed in its own folder to easily track datasets.

To install ERPEEG:

1. Copy the .mlappinstall file to your current working folder in MATLAB.
2. Right click on the file and select 'Install'.



Dataset Loading and Setup

ERPEEG is initialized by selecting the app from the MATLAB apps toolbar. Users can also open the tool by running the *erpeeg_main* function directly in the MATLAB command window (to do this ensure the software folder, under MATLAB -> APPS, has been added to the path). Running the program will display the main GUI of ERPEEG.





Figure 1: ERPEEG Main GUI

To load a dataset:

1. Select the **Working Folder** button, calling a 'select working folder' window. Using the pathing tool, select the folder you have created containing your base .set file
2. After selecting the folder, select the **Dataset** button and choose your .set file in the working folder

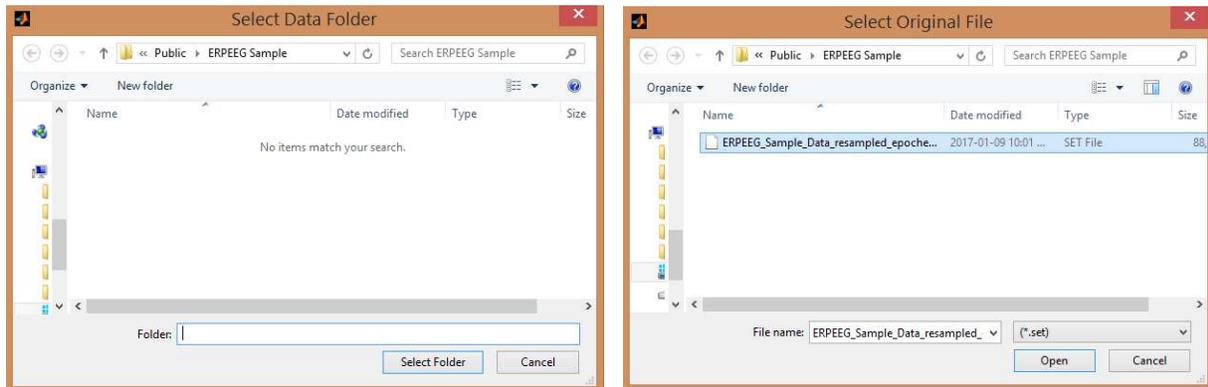


Figure 2: Selecting folder (Left) and File (Right)

If you have previously processed the dataset in the working folder, the ERPEEG GUI will indicated a completed dataset by colouring the step green (see **Figure 3** below)

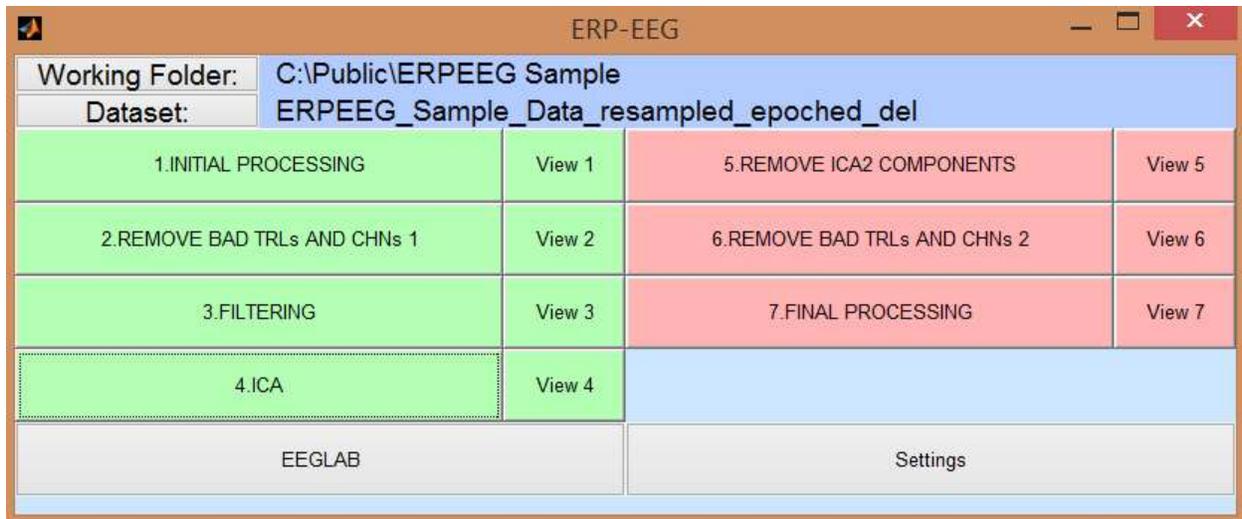


Figure 3: Main GUI displaying completed steps (in green) and settings menu

The **Settings** button in the main GUI brings up a smaller window, allowing the user to select and change the processing parameters of the toolbox. Users can also access main GUI of EEGLAB main GUI at any time during data processing through the **EEGLAB** button.

Settings

Parameters for data processing can be changed in the settings tab. Settings are currently available for:

- Step 1 – Initial processing
- Step 2 – Removal of bad trials and channels
- Step 4 – ICA
- Step 5 – ICA Component Removal
- Step 6 – Removal of bad trials and channels 2
- View Data – Graphing options for View Data feature at each step

Due to the sequential nature of the ERPEEG workflow, if the settings are changed for a specified step, the workflow will reset to that step (i.e., all data processing from the specified step and onwards will be erased). We highly recommend users to specify all parameters in the setting tab before processing the data to avoid unnecessary resets of the processing workflow.

Step I – Initial Processing

The user must epoch the data prior to using ERPEEG. Future versions of ERPEEG will include an option for epoching. After loading the data, select the **Initial Processing** button on the main GUI to initiate the first processing step, which guides the user with a series of pop-ups through the following processes:

1. Baseline Removal

Baseline removal can be performed based on a user-selected time range (default is set relative to the selected pre-stimulus range).



Figure 4: Baseline Process

2. Resampling

Resampling is recommended to improve toolbox performance and reduce data storage requirements. ERPEEG will prompt the user to resample through the following window:

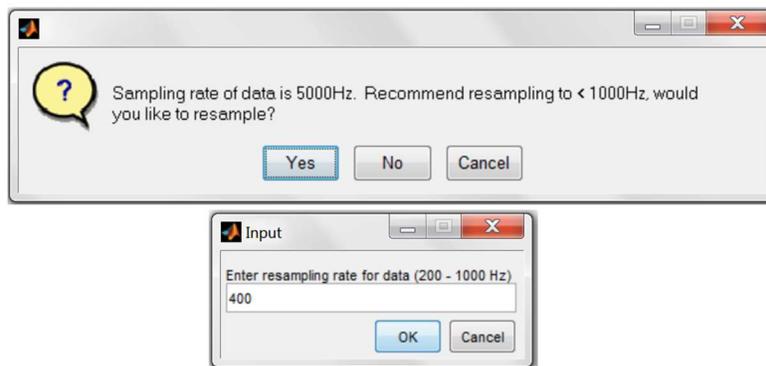


Figure 5: Resample Prompt and Input

3. Channel location lookup

After resampling, the user is prompted to select a channel location file using EEGLAB's `pop_chanedit()` function or load default locations using a database of 385 defined channel labels from the 'Standard-10-5-Cap385.elp' file originally from the EEGLAB distribution.

If the user decides to keep channels without channel locations, a warning is provided. It is highly recommended to delete channels without channel locations in Step 1 or alternatively in Step 2. Channels without any location information cannot be interpolated and therefore may cause issues if trials within these channels are marked for deletion. Non-EEG channels should also be removed before ICA. Future versions will allow the user to select channels for ICA.

An automatic pop-up window shows the channel locations (if available) in a topographic display, with a scrolling list for the user to select channels for removal. If the channel locations cannot be found, a pop-up window will alert the user.

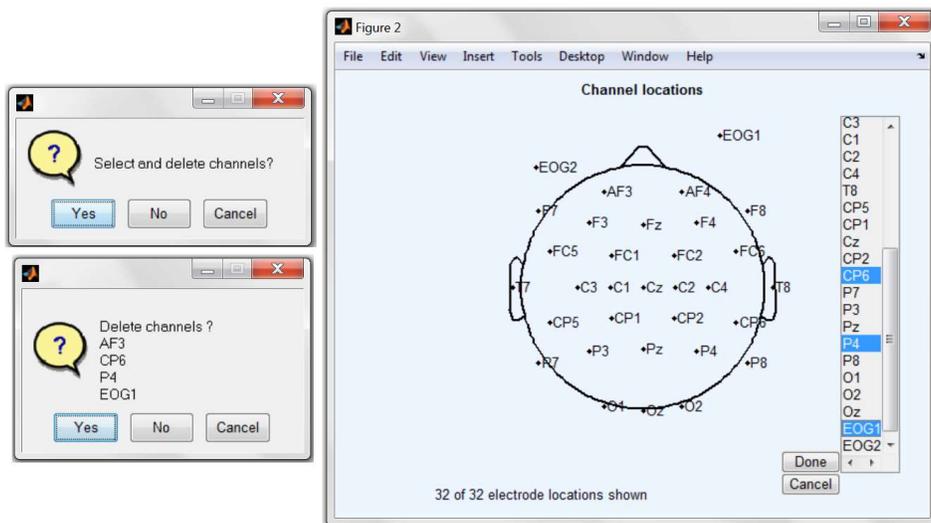


Figure 6: Channel Removal Display

Step II – Removal of Bad Trials and Channels

1. Select Attribute for Display

ERPEEG displays data through the **Plot Trials** and **Plot Channels** GUI with a user-selected attribute. In the Plot Trials display, trials are displayed based on the chosen attribute extracted over all channels. In the Plot Channels option, a subplot is generated for each channel to display the **ATTRIBUTE** value at each trial for the specified channel.

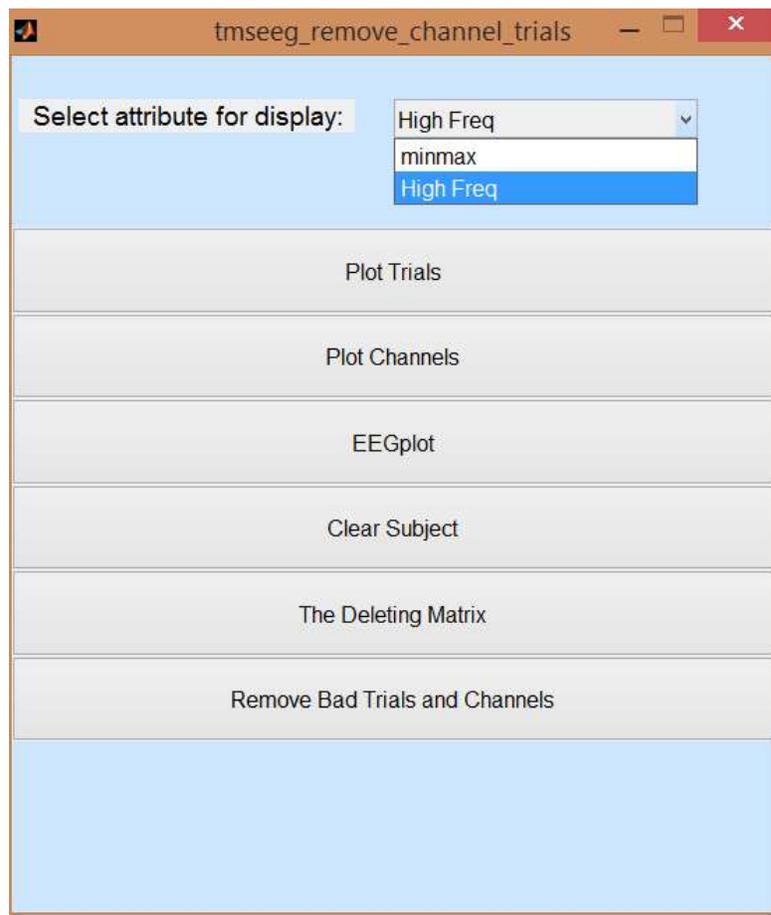


Figure 7: Main GUI for Trial/Channel Removal with ATTRIBUTE Selection

2. Plot Trials

Plot Trial uses the selected **ATTRIBUTE** to calculate an attribute value for each trial. This is displayed through a scatterplot (**Figure 8**). By clicking these dots, the relevant trial data is displayed through a new GUI, allowing the user to delete selected channels in the trial, or the whole trial.

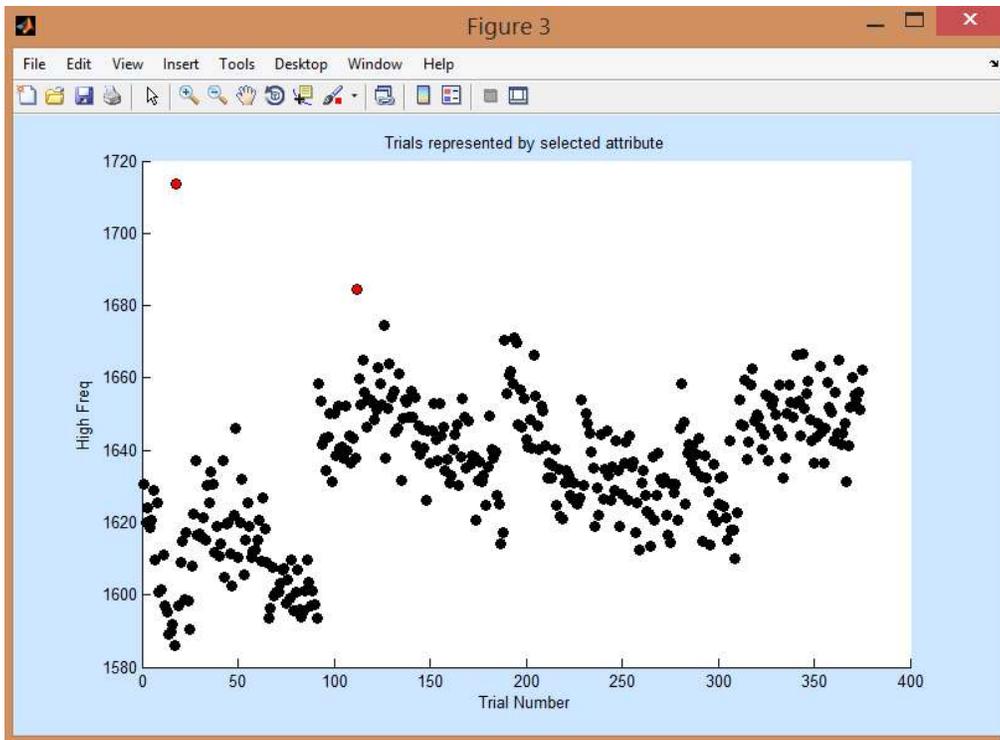


Figure 8: Plot Trials GUI – Trials that have been marked for deletion are shown in red. Clicking a dot with the mouse brings up the display shown in Figure 11 below.

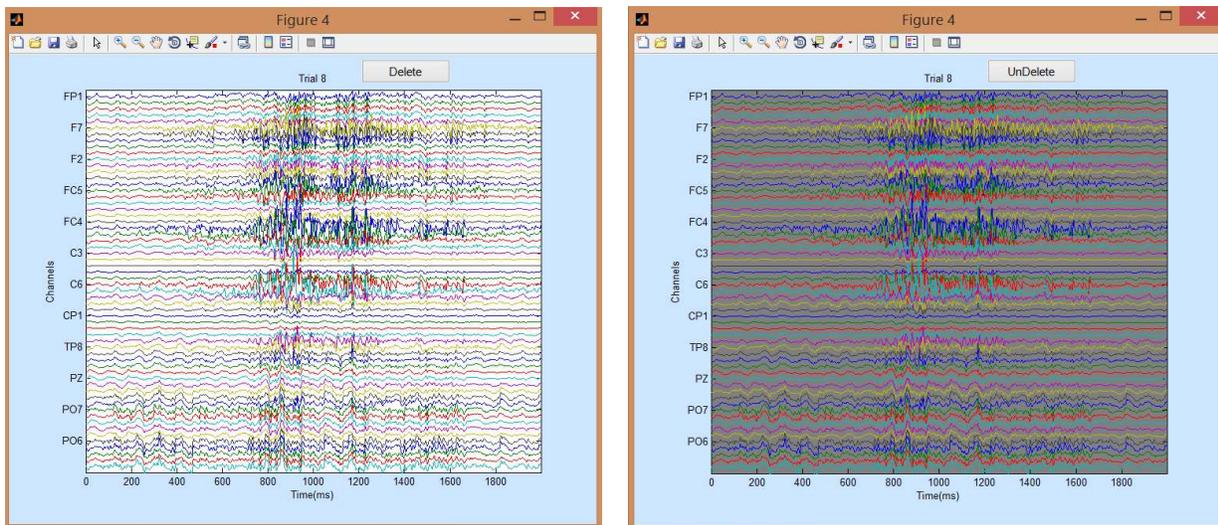


Figure 9: Trial Deletion GUI – Called by clicking on the corresponding dot in the Plot Trials GUI. Channel data is displayed for the selected trial (in this case trial 8). Right window shows the trial as deleted.

In this **Trial Deletion GUI (Figure 9)**, channels can be selected for deletion for the duration of the trial using the **Delete** button. Alternatively, if the user clicks on the **Delete** button without selecting any channels then the whole trial is deleted.

3. Plot Channels

NOTE: Currently the Plot Channels GUI can only support data files with a maximum of 264 epochs. This will be extended in future versions of the software. For larger file sizes, users may want to use the other functionalities (Plot Trials (above) or EEGPLOT (see below)) or split the file into smaller subsets.

ERPEEG offers easy visualization so that the user can assess data for removal with the **Plot Channels** GUI. Plot Channels creates a subplot for each channel in the dataset as seen in **Figure 10**. This scatterplot displays the ATTRIBUTE value of each trial within the channel (**Figure 10**). Selecting these subplots will bring up the **Channel Removal GUI** (**Figures 11-12**), while selecting the dots in the subplots will bring up the **plot Trial GUI** as seen as **Figure 8** for that channel from where you can access the **Trial Deletion GUI** as seen in **Figure 9** by selecting again one of the dots.

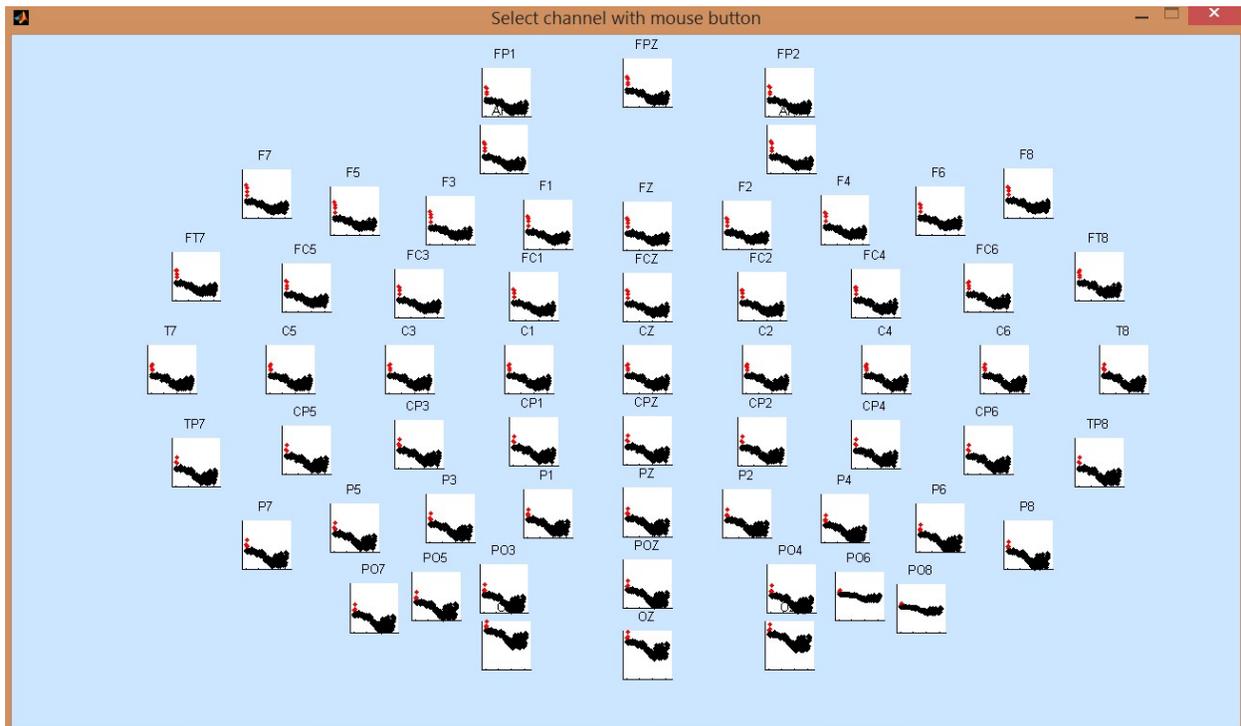


Figure 10: Plot Channels GUI – The position of the scatterplot for each channel is based on the loaded channel co-ordinates file. Dots in each channel subplot indicate the ATTRIBUTE value of each trial, and a blacked-out channel indicates that it has been set for removal.

IMPORTANT NOTE: Users are highly recommended to delete channels without co-ordinates at this step. Alternatively, if users want to retain these channels they can make sure they do not delete any trials within these channels. Deleted trials within channels will be interpolated (if below the user-specified threshold) at the end of step 2 and channels with no channel co-ordinates cannot be interpolated. It will result in an error.

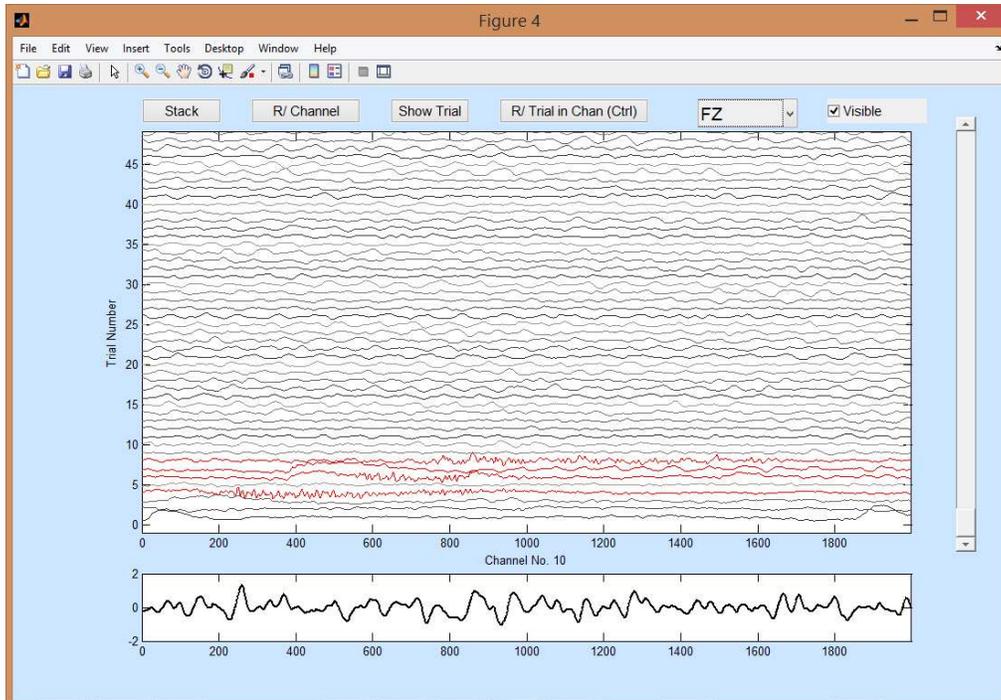


Figure 11: Channel Removal GUI – Spread Mode

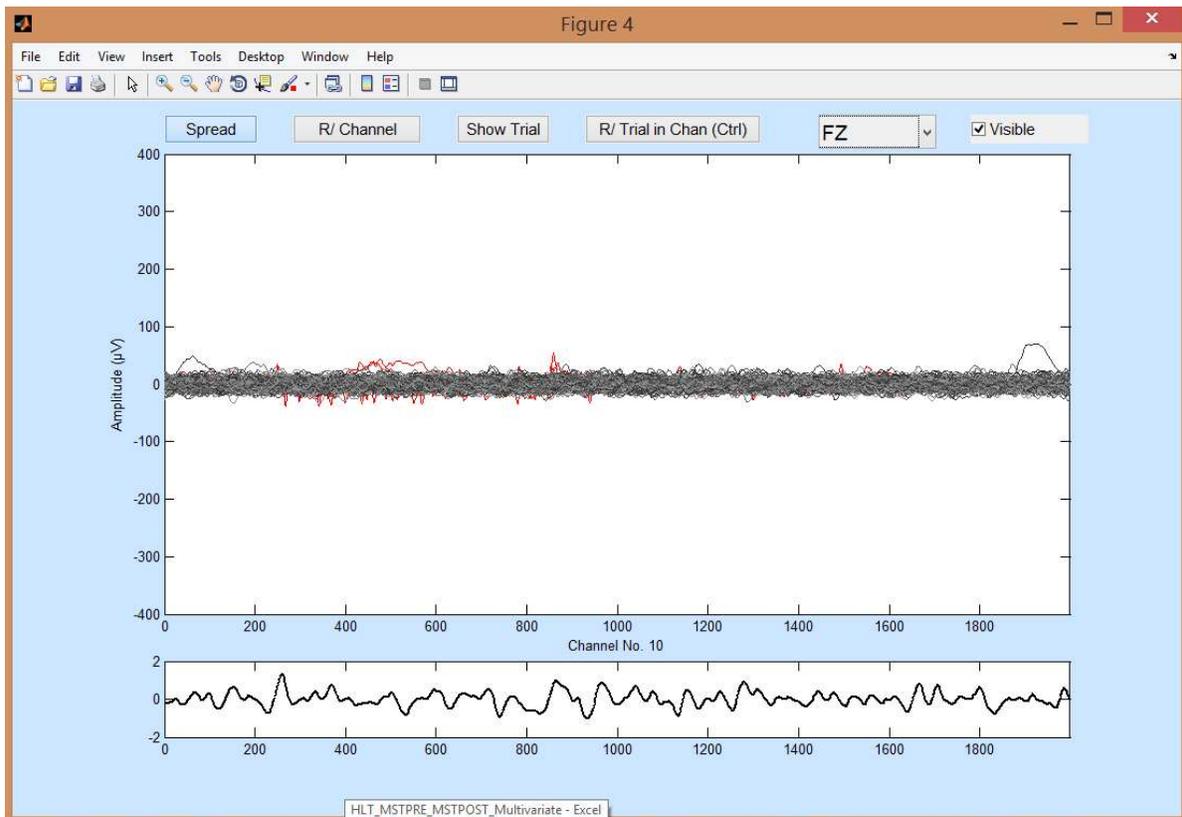


Figure 12: Channel Removal GUI – Stacked Mode

Figure 11 and **Figure 12** show the Channel Removal GUI in its two display modes: Stacked and Spread. The spread display plots the channels in a more traditional offset format, while the stacked display plots all traces on the same baseline for comparison. Three options exist for deleting data in this window:

- Removal of Trial in Channel** – Delete an individual trial-channel pair, by right-clicking the data segment or by selecting the data segment and clicking the **“R/ Trial in Channel”** button.
- Remove Channel** – Using the **“R/ Channel”** option, mark all trials in the channel for deletion.
- Trial Deletion** – Select a trial and use the **“Show Trial”** button to open the Trial Deletion GUI to delete the entire trials or selected channels within the trial (**Figure 9**).

Trials marked for deletion will be seen as red in the Channel Removal GUI if the “Visible” option is selected. The channel drop down menu allows quick scrolling through the channels and can be operated using the up and down arrows on the keyboard.

4. EEGPLOT

For users familiar with the *eeplot* function in EEGLAB, we have included this function directly linked to ERPEEG:

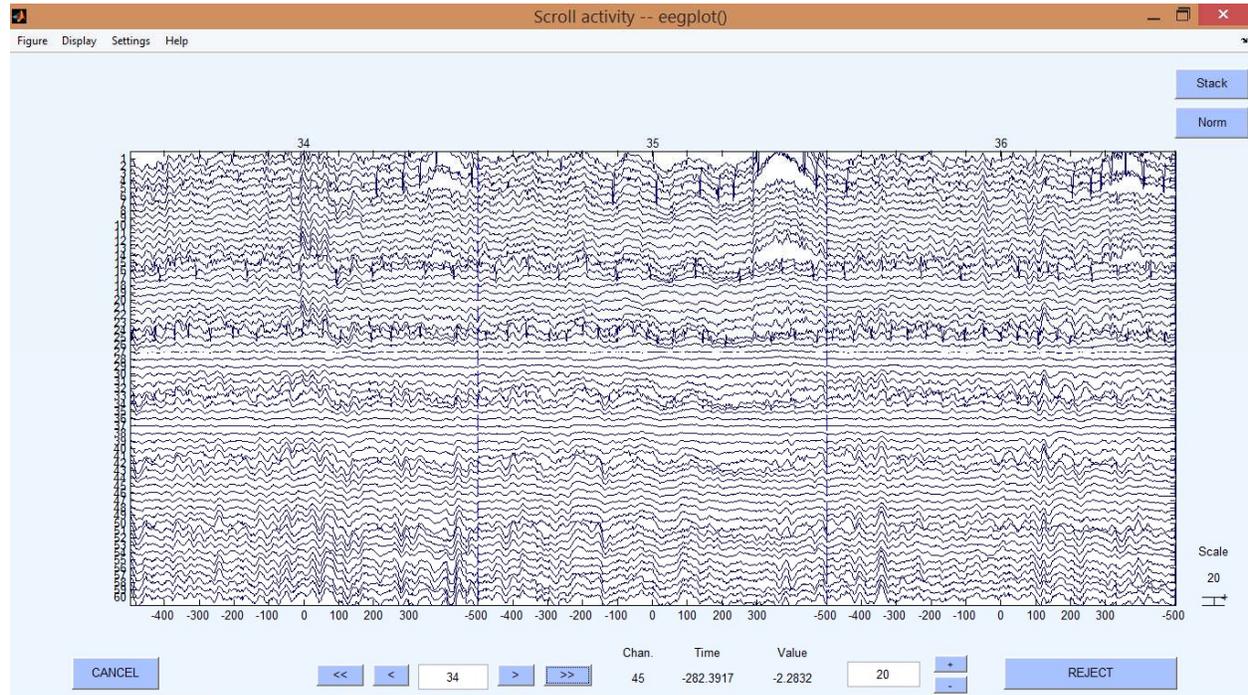


Figure 13 – EEGPlot Option – Called using the EEGplot option from the remove trials and channels GUI. Selected trials (highlighted in blue) are removed by pressing the REJECT button.

5. Clear Subject

If the user needs to restart the cleaning process, the trials/channels marked for deletion can be quickly cleared using the **Clear Subject** button.

6. Deletion Matrix

The **Deletion Matrix** button offers an efficient quality control on the deletion process by plotting the trials and channels currently selected for deletion in a matrix (Trials vs Channels) format:

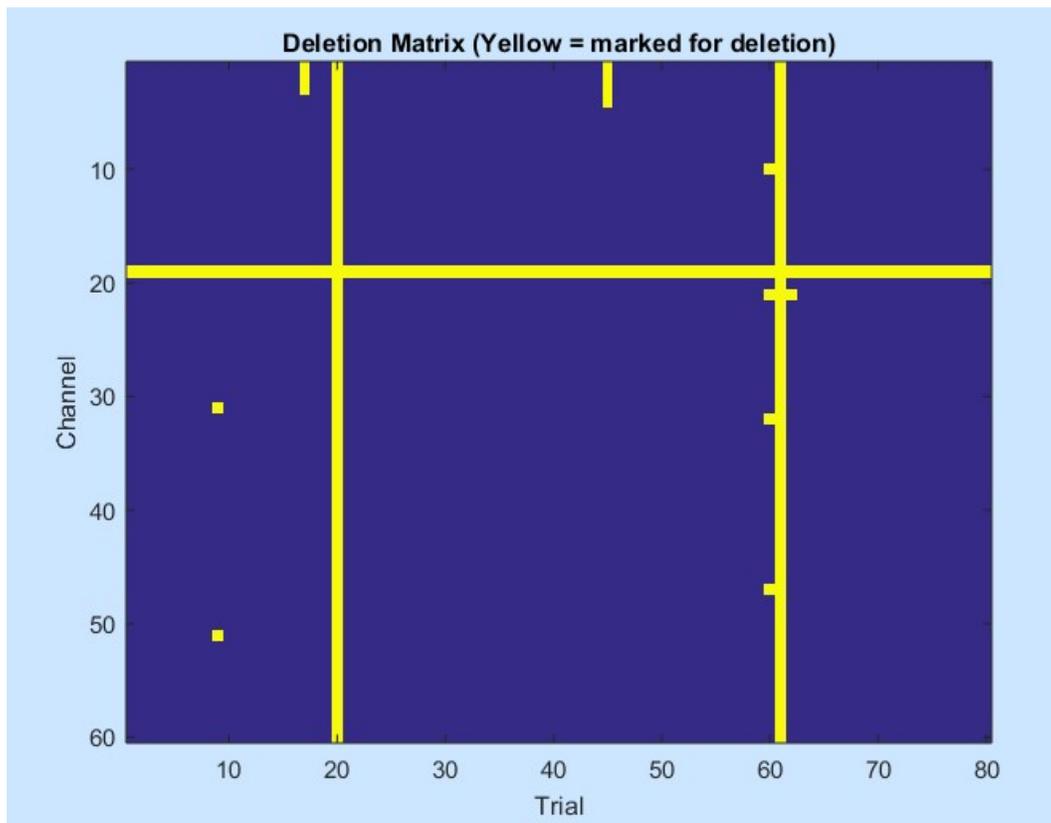


Figure 14: The Deletion Matrix

Step III – Filtering

ERPEEG currently offers two options for data filtering:

- **FIR filtering** – filtering with a basic FIR filter (recommended 1-55Hz) to remove unwanted frequencies and avoid power line noise. The user can adjust the filter order using the filter order button.
- **IIR Filtering** – Bandpass filtering using a butterworth filter, with a notch filter centred at 60Hz for removal of power line noise. Currently, the user can adjust the frequency bands of the filter, and the order of the butterworth filter by selecting the filter order button.

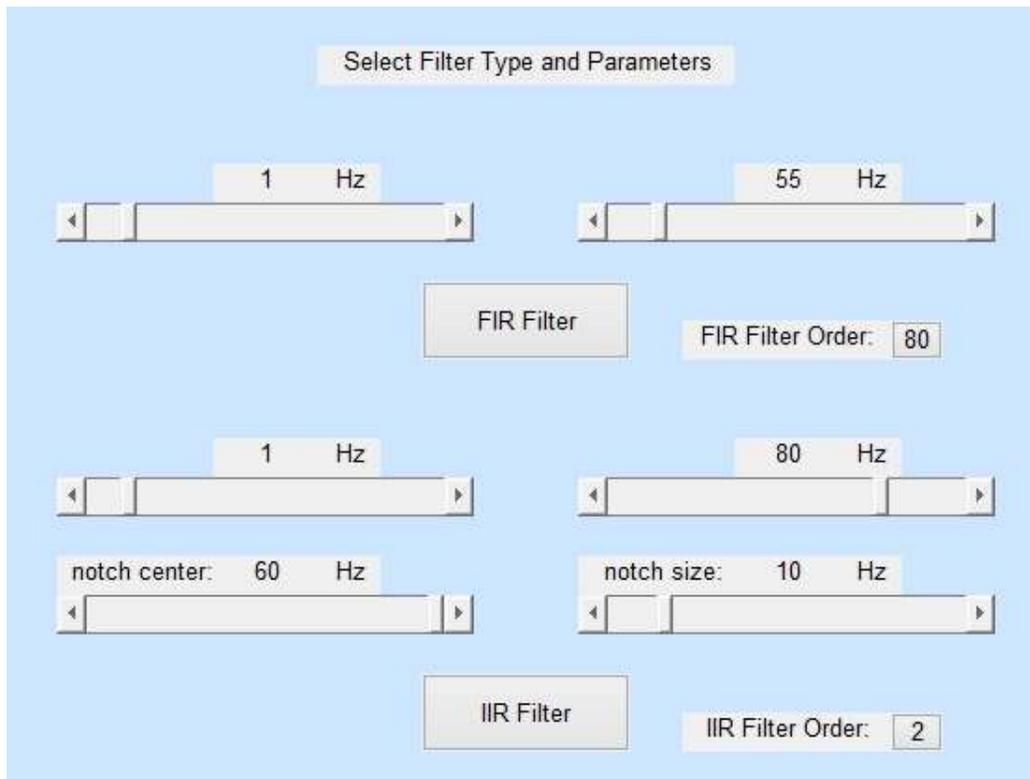


Figure 15: Filtering GUI

Step IV – ICA

ERPEEG uses the *pop_runica* function from EEGLAB to run ICA, currently using the **FASTICA** algorithm (see introduction for downloading link). Assuming FASTICA has been downloaded and is on the MATLAB path, selecting the ICA button will run ICA on the latest dataset to extract the components. The maximum number of components that can be extracted is equal to the number of channels in the dataset. The number of components extracted can also be specified in Step 4 settings (as a percentage of remaining channels; 50% recommended for resting EEG data). Users can specify the channels that should be used with ICA. If non-EEG channels are present in the data, please unselect them before ICA.

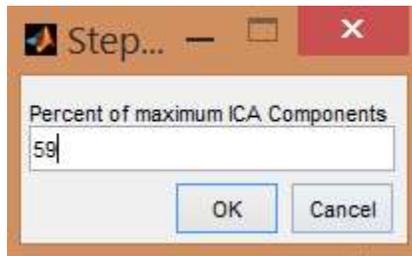


Figure 16: Step 4 Settings

Step V – Remove Artifacts with ICA

Using the calculated ICA components in step 4, step 5 calls the component removal GUI to facilitate tagging and removal of common EEG artifacts:

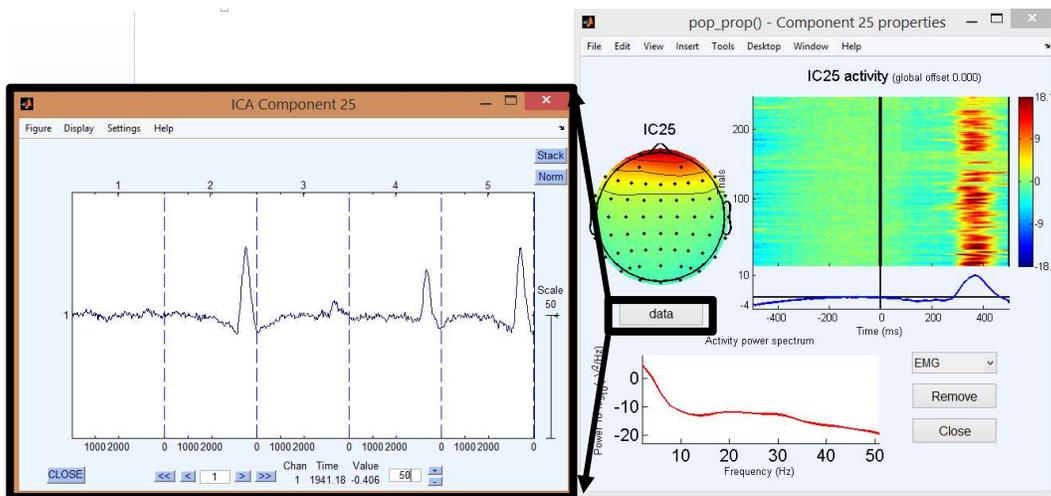
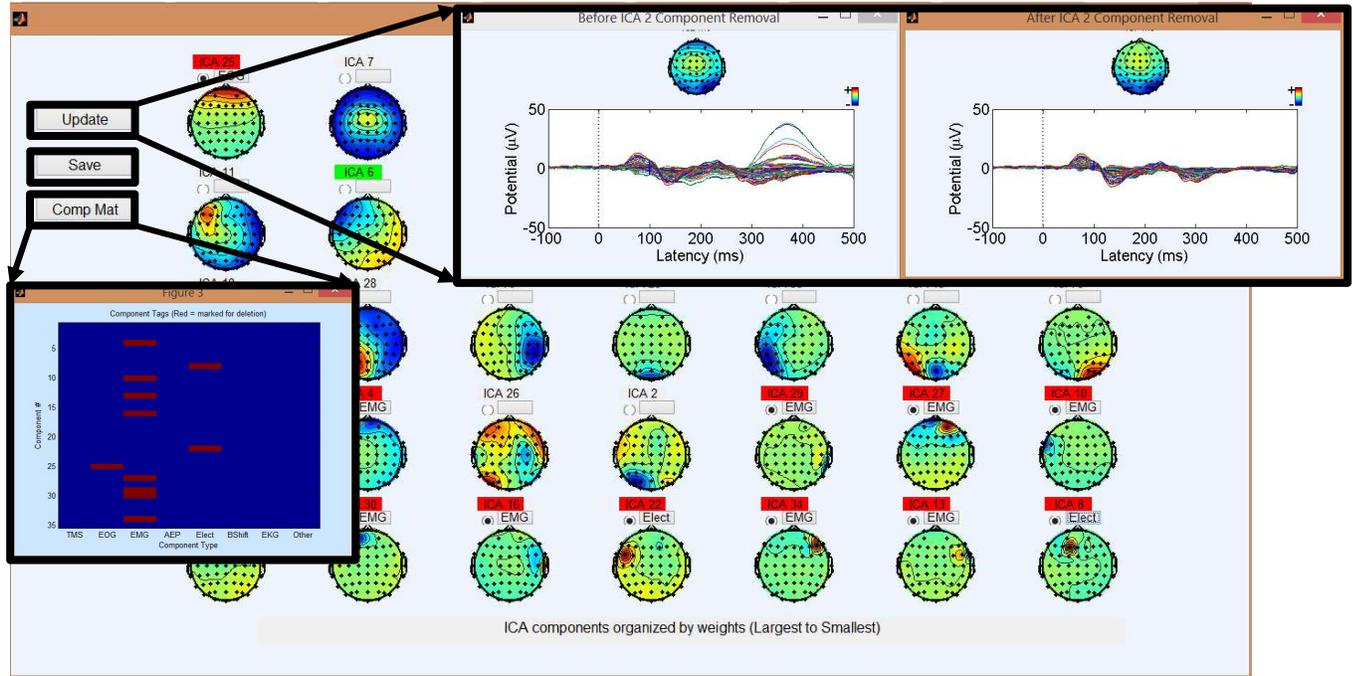


Figure 17: ICA Component Removal Display

The main component removal GUI shows topographic plots of the ICA components, sorted in decreasing order of magnitude. Selecting the button of the relevant ICA component brings up a custom display showing information on topography, frequency content, and trial/channel amplitude information (**Figure 17**). Using this display, components can be marked using the drop-down menu as a specific artifact type or viewed more closely using the **'data'** button. Once a component is marked as an artifact for deletion,

the ICA component label turns red, while viewing (not marking) a component will mark the label as green. In the top-left corner of the GUI, the user can view the butterfly waveform before and after the removal of the marked components by clicking on the **Update** button. However, these marked components are not removed until the user decides to **Save** the workspace. Finally, for quality control, the user can visualize the number of deleted components and the type of artifacts deleted through the **Comp Mat** button.

Step VI – Remove Channels and Trials 2

Step 6 includes a second round of trial and channel removal, targeting any residual noise or artifacts introduced in the processing workflow. Please refer to the Step 2 instructions as the noise removal process is identical. The parameters of this step can be modified as needed through the **Settings** button in the main GUI of ERPEG.

Step VII – Final Processing

ERPEG performs three tasks during the final processing step: (i) interpolation of channels deleted in Steps 2 and 6, and (ii) re-referencing of channel data (optional). Interpolation is performed using the *pop_interp* function from EEGLAB. By default, the function is specified to use the spherical method. This step concludes the processing of EEG data from raw signals to clean ERPs. At any given step during the workflow, users can visualize the progressive removal of artifacts through the averaged ERP butterfly plot by clicking on the "**View Step**" button in the main GUI (**Figure 18**) or continue data processing in EEGLAB using the **EEGLAB** button.

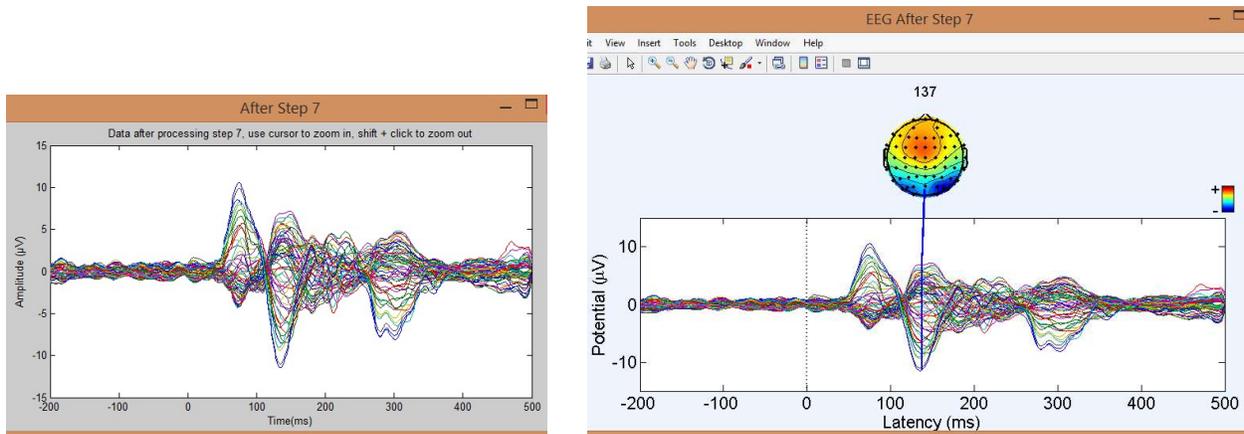


Figure 18: Butterfly plots shown when selecting the View Step button (with and without topographical map)